CONF-8503179--1

**LA-UR** -85-1094

LA-UR--85-1094

DE85 009602

TITLE: ON THE CHARACTERISTICS OF A NUMERICAL
FLUID DYNAMICS SIMULATOR

AUTHOR(S): Karl-Heinz A. Winkler, Michael L. Norman,
and John L. Norman

SUBMITTED TO: Austin Symposium on Algorithms,
Architectures, and the Future of
Scientific Computation
March 18-20, 1985

Los Alamos Los Alamos National Laboratory
Los Alamos, New Mexico 87545

# On the Characteristics of a Numerical Fluid Dynamics Simulator[*]

Karl-Heinz A. Winkler, Michael L. Norman,
and John L. Norton
Applied Theoretical Physics Division
Los Alamos National Laboratory

[*] Work performed under the auspices of the U.S. Department of Energy.

## DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

ABSTRACT

John von Neumann envisioned scientists and mathematicians analyzing and controlling their numerical experiments on nonlinear dynamic systems interactively[1]. We describe our concept of a real-time Numerical Fluid Dynamics Simulator NFDS, and derive its characteristics adopting the following as guiding principles:

a) that the performance characteristics (information output, throughput and storage) of the NFDS be defined by the maximum rate at which the researcher can absorb the results of his / her simulation, thus establishing an impedance match between man and machine,

b) that the NFDS off-load from the researcher's brain in every way possible the routine tasks of data analysis that can be done automatically, and

c) that the NFDS is operated as a dedicated experimental device, much as a wind tunnel, and that the researcher has complete control over the apparatus and experiment.

We envision the NFDS to be composed of simulation processors, data storage devices, and image processing devices of extremely high power and capacity, interconnected by very high throughput communication channels. We present individual component performance requirements for both real-time and playback operating modes of the NFDS, using problems of current interest in fluid dynamics as examples. Scaling relations are derived showing the dependence of system requirements on the dimensionality and complexity of the numerical model. We conclude by extending our analysis

to the system requirements posed in modeling the more involved physics of radiation hydrodynamics.

## JOHN VON NEUMANN'S VISION

In a remarkable paper[1] that was never published during his lifetime, John von Neumann developed in 1946 the concept of a digital computer, computational synergetics, numerical experiments, and software productivity. The von Neumann machine architecture, consisting of a central processing unit, memory, logical control, and an input-output system has essentially dominated the basic design of digital computers for the past 40 years. Only recently, with the appearance of parallel processing machines, has the development gone beyond the basic von Neumann concept. His notion of computational synergetics, i.e. the use of numerical simulations for "providing us with those heuristic hints which are needed in all parts of mathematics for genuine progress", has received particular attention in the literature, see e.g. the review papers by Birkhoff[2] and Zabusky[3]. Software productivity is of ever increasing importance as computational tasks grow in complexity. Von Neumann's statement that "every effort must be made to simplify the coding of problems" is as true in 1985 as it has been in 1945, may be even more so. This issue is finally being taken seriously through the development of new programming languages such as ADA or GIBBS.

One defining characteristic of a numerical experiment in von Neumann's sense is the ability of the human researcher "to exercise his intuitive judgement as the calculation develops". This necessitates the capability of the computer system to provide the user with a continuous read out of all essential information "while the calculation is in progress". The user "can then intervene whenever he sees fit". Consequently, von Neumann places enormous emphasis on what he calls the input-output organ. Its purpose is to present final results, generate restart dumps for later usage, and let the user follow the computation by providing intermediate results of the simulation and error diagnostics in graphical form. Of course, von Neumann was thinking in terms of the technological base of his time, and therefore discussed the use of oscilloscopes as the primary graphical output medium for intermediate results and error diagnostics.

In this paper we want to undertake the task of translating von Neumann's original vision into our modern technological context. We will generalize his ideas of graphical error diagnostics by introducing the powerful tools of image processing and enhancement, and pattern recognition into the intermediate and final analysis of numerical experiments. Finally, we will give a system analysis that will show the computational and communicational burden that high resolution continuum mechanics simulations, such as multi dimensional gas dynamics and radiation hydro computations, place on any given computer environment.

## IMPEDANCE MATCH MAN-MACHINE: GIGABAUD

Productive real-time numerical experiments must establish an impedance match of information flow between man and machine. That means in particular that the information flow of the numerical experiment has to match the human needs, not the other way around. Otherwise, the productivity of the user of such a facility will go way down because of excessive waiting periods at the user's end. In this context the relevant question we therefore have to address is: What is the maximum baudrate with which the human brain-eye system can be driven? An estimate of this quantity, correct to within an order of magnitude, gives: GIGABAUD, i.e. gigabits per second.

We have arrived at this estimate, which follows alone from human physiology, in the following way. The spatial resolution of the human eye is about 1 minute of arc, its approximate field of view 60 × 60 square degrees, color sensitivity varies throughout the visible spectrum, but stays within 8 bits in the red, green, and blue channel, leading to a maximum of 24 bits of color resolution. One also has to consider the fact that the eye is somewhat more sensitive in grey shades than in color. If we couple these numbers with a display rate of a mere 3 Hz we already arrive at the fact that the human physiology allows us to receive visual information at a GIGABAUD rate. Realistically speaking the rate is a few GIGABAUD because our eye operates on a higher than 3 Hz refresh rate.

Good raster graphics devices operate on a 60 to 70 Hz noninterlaced refresh rate, avoiding the detection of flickering on the screen by the eye. This does not mean that we actually could drive our eye at that much higher baudrate as would be indicated by the use of 70 Hz. In fact, typically the entire information content of an image does not change completely from frame to frame. Instead, there will be a lot of redundancy in neighboring frames, which we inadvertently exploit by using our brain as a differential analyze . So, the basic fact remains: We can receive visual information at GIGABAUD rates!

As has already been pointed out by von Neumann, these high baudrates will be followed by a much slower human thought process. This slow process of making sense of and understanding the numerical simulation, differing substantially in speed from user to user, hopefully will transform the flow of information into knowledge and finally insight. As we all know, the result of a simulation should be insight not numbers (Hamming).

## IMAGE PROCESSING AND ANALYSIS

Thus, we have established the need for an efficient GIGABAUD-interface between the user and the machine. Such an interface is certainly necessary but in many cases not sufficient to allow a creative use of algorithmic developments in the solution of nonlinear physical problems. A missing element is the ability to process the images that result from complex calculations, such as coupled systems of nonlinear PDEs. Analyzing the numerical data AS IMAGES with the best tools available will allow us to recognize patterns, often of an unexpected nature. Such techniques have been well-developed in the context of the interpretation of observational and experimental data. The rationale and procedure for doing such analysis applies equally well to the results of numerical experiments on digital computers. The bottom line is: once you have a billion numbers, it doesn't matter any more how you got them, whether from observation, laboratory or numerical experiment. You have to analyze them, preferably in image form, in order to understand their physical significance.

Once an image has been generated, for example as a bit map for a video display screen, it can be processed to highlight patterns not obvious from the "raw" picture. The combination of the human mind and eye is without doubt the best pattern recognizer. However, they can be greatly aided by image processing software. A common but valuable usage in the analysis of experimental data is image enhancement through digital filters. This has been dramatically illustrated in the Jet Propulsion Lab pictures of Jupiter and its satellites. Such techniques could be similarly used to emphasize discontinuities in gas flows that result from physical phenomena such as shock fronts and contact discontinuities. Another example is the extraction of both the most and least significant parts of the data for separate treatment. The most significant bits hopefully will contain the essence of the physics; the least significant bits surely will indicate the numerical noise level of the experiment. They also may indicate the development of certain trends which otherwise would stay unnoticed because they are hidden in the full solution through the presence of well developed dominant features. As a final example for the use of image analysis methods let us consider the comparison of images that result from different calculational schemes but the same physical problem. In this case the comparison will show us the sensitivity of the solution due to different numerical methods, grid resolution or refinement.

Even more powerful will be the use of these tools to give the necessary feedback for the control and optimization of the numerical experiment itself. This kind of feedback can be provided at three increasing levels of complexity.

The first is that, as a result of the image processing and analysis, it is possible to improve on the algorithmic part of the calculation, e.g. mesh refinement. The information can be used to modify the details of the calculation to improve the results the next time around. This is the most common way such improvements take place today, albeit often based on calculations long since done and using the human mind-eye combination as the image processor for studying a paper plot.

Another increase in user productivity will be achieved by allowing interactive feedback of image processing information (through a user at a graphics station) to affect the course of a running simulation. For example, adaptive mesh techniques often involve the setting of certain global parameters that determine what parts of the physics are best resolved. As the global parameters enter in a highly nonlinear manner, their effect on the overall system cannot easily be predicted. If these parameters were accessible, selectively, through I/O devices such as joysticks, the user could instantaneously improve (or attempt to improve) a running simulation and receive immediate information about the effect of such changes. It is quite likely that the intuition developed from such computing sessions would lead rather rapidly to more general methods and certainly to a better tailoring of existing algorithms to problems of interest.

At the final level of effectiveness, the image analysis is directly coupled into the running problem in the same way as control and optimization techniques are used in an industrial setting. The user is now really in the driver's seat of the numerical experiment and will be able to eliminate a lot of waste. In a way, his relation to the numerical experiment is much the same as that of a pilot to a flight simulator. The example of adaptive mesh algorithms is also applicable here. As methods of optimizing the parameters result from experience with running a large number of problems, it is just a matter of time until connections can be established between image analysis and automatic optimal settings of these parameters.

A key point to make is that as the techniques of image processing and analysis become available at the three levels described above, they will evolve in ways that cannot be predicted. The framework of a flexible, interactive, graphics interface between the user and the machine will be crucial in the course of this evolution. The possibility of improving the productivity of algorithm development and problem solving by many orders of magnitude, particularly on the human timescale, seems within our grasp.

## INTEGRATED SYSTEM REQUIREMENTS

Having estimated a reasonable upper limit on human throughput $T_h$, we can now address the following question: What are the requisite system capabilities (execution speed, network throughput, and storage capacity) such that for an arbitrary application, user productivity is maximized? In other words, how fast must the simulation processors and communication channels be when operating the NFDS in real-time mode and how large must the fast- access memory be when operating in play-back mode, such that the output of the numerical simulation is conveyed to the user's brain-eye system at a sustained rate $T_h$? These numbers will now be derived for two specific applications with the help of a few simple relations.

The output of the simulation processors O in words per second is given by

$$O = X / \bar{C} \qquad (1)$$

where X is the processors' aggregate execution speed in equivalent operations per second, and $\bar{C}$ is the application algorithm's average complexity per zone measured in number of operations per updated field variable. Factors affecting $\bar{C}$ are the size, dimensionality, and degree of coupling of the physical system (i.e., differential, integral, integro-differential), which will be reflected in the numerical method of solution. $\bar{C}$ will depend on the algorithm's order of accuracy and number of unknowns in different ways for explicit and implicit systems, and whether direct or iterative methods are used for the solution of the latter.

The simulation processors can be thought of as an information pump generating O words/second of raw data. We would like to tap that output and route it to image devices that display information in any desired form as fast as it is being produced. Exploiting the redundancy in the raw data, we can write the inter-system throughput baud requirement $T_{is}$ as

$$T_{is} = W\ O\ f_d\ f_c\ f_s\ f_r\ \min(P/S,1) \qquad (2)$$

where W is the word length in bits, and the f's are redundancy and reduction factors defined as follows: $f_d$ is the dump frequency factor equal to the inverse of the interval in cycles between dumps; $f_c$ is the data compression factor representing the truncation of a full machine word of length W to of order 24 bits of color information; thus, $f_c = 24/W$; $f_s$ is the selection factor, reflecting the fraction of the total number of field variables you would like to display at once; and $f_r$ is the redundancy factor reflecting the fractional change of the image in time. Estimates for these factors are given below; their product can be several orders of magnitude smaller than unity, significantly reducing inter-system throughput requirements below the raw output baudrate W × O. The last factor in equ. (2) reduces $T_{is}$ by a factor P/S if the problem size S in zones exceeds the number of pixels P on the image device.

$T_{is}$ as defined in equation (2) represents the rate at which nonredundant data is drawn from the simulation processors to be displayed by an image device, and defines the rate the intersystem communication channels must sustain. The internal throughput of the image device $T_{id}$ must be greater than $T_{is}$ by a factor $1/f_r$ because the nonredundant data must be expanded to fill the screen.

Our principle of impedance matching man and machine implies that $T_{id}$ must equal the effective human throughput $T_{h_{eff}}$:

$$T_{id} = T_{is} \;/\; f_r = T_{h_{eff}} = \min(S/P,1)\; T_h \qquad (3)$$

$T_{h_{eff}}$ is less than the maximum human throughput $T_h$ by a factor S/P for $S < P$ simply because a deficit in information due to low resolution or restricted field of view cannot be compensated for by a higher display rate.

Substituting for $T_{is}$ from equ. (2) into equ. (3), eliminating O via equ. (1), and solving for X we find:

$$X = T_h\, \overline{C}\, (S/P) \;/\; (W\, f_d\, f_c\, f_s) \qquad (4)$$

This relation specifies the execution speed X required to maintain a throughput $T_{h_{eff}}$ to the user for a simulation of complexity $\overline{C}$, taking into account reduction factors. Note that X scales linearly with the problem size S, as it should.

Let us illustrate the utility of equ. (4) with two examples of current interest: 3-D explicit gas dynamics and 1-D implicit radiation hydrodynamics. The complexity factors $\bar{C}$ are evaluated and tabulated below on the basis of measured performance of the two fully vectorized codes PPM[4] and WH80s[5], respectively, on a Cray-1 computer assuming a sustained execution rate X of 20 Megaflops. V is the number of variables per zone and R the execution rate of the code in updated points per second.

Case 1: Suppose we want to perform a 3-D explicit gas dynamics computation on a grid of 1,000,000 zones and monitor the outcome in real time while sampling only every tenth timestep. Assuming $\bar{C}$ scales linearly with the dimensionality of the system, we derive a value of $\bar{C} = 300$ on the basis of Table 1. Assuming $W = 64$, $f_d = 1/10$, $f_c = 3/8$, $f_s = 1/5$ and $S/P = 10^6/3600^2$, we find $X = 48$ gigaflops. Such a machine could sustain an effective baudrate of 1 gigabaud × S/P = 77 megabaud of image information from the above application to the user. The use of a comfortable 10,000,000 zones makes $X \approx 0.5$ teraflop and the output baudrate approaches $T_h$.

Case 2: Suppose we want to compute 1-D implicit radiation hydrodynamics flows including the full space-angle-frequency coupling of the radiation field on a fully adaptive mesh in

Table 1: Sample complexity factors $\bar{C}$

| Code | Description | V | R | $\bar{C} = X/RV$ |
|---|---|---|---|---|
| PPM | 2-D explicit gas dynamics | 4 | 25,000 | 200 |
| WH80s | 1-D implicit radiation hydro | 8 | 1,000 | 2500 |

space, angle and frequency as proposed in Winkler, et al.[6]. Suppose we use 50 x 50 zones in the angle-frequency mesh, and 300 zones in the spatial coordinate. This can be accomplished within the context of the WH80s code by including the 4 × 50 × 50 intensities and mesh coordinates as additional unknowns that must be solved for at every grid point, increasing the number of unknowns per spatial zone by a factor (10000 + 8) / 8 ≈ 1000. Since $\overline{C}$ is defined on a per variable basis, it scales as the square of this factor if direct methods such as block gaussian elimination are used to solve the implicit system. Assuming $W = 64$, $f_d = 1/2$, $f_c = 3/8$, $f_s = 1/8$ and $S/P = (300 \times 50^2)/3600^2$, we find $X \approx 3 \times 10^{16}$ flops. This enormous rate indicates that direct methods are impractical for solving this problem. Iterative techniques, such as ILUCG and multigrid, seem to be the only hope for such problems in the near future, since $\overline{C}$ typically scales only weakly with the number of unknowns. It remains to be seen whether such techniques can handle the typically ill-conditioned matrices of the linear systems involved. However, assuming iterative techniques can be made to work on such problems and that $\overline{C}$ is of order 2500 independent of the number of unknowns, then we find $X \approx 100$ gigaflops.

Thus, we can conceive of running these two applications and monitoring their progress in real time on a Numerical Fluid Dynamics Simulator consisting of simulation processors delivering tens of gigaflops and communication channels carrying image data at gigabaud rates to extremely powerful image processing devices. In playback mode, no demand is made on the simulation processors. Rather, the data is written to storage at a rate determined only by the speed of the processors, and stored for later playback at gigabaud rates. The requisite storage ranges between 10 gigawords and 1 teraword depending on what fraction of the total data is stored.

Acknowledgment: We gratefully acknowledge the assistance of Stephen W. Hodson in preparing this manuscript.

REFERENCES

1. H. Goldstine and J. von Neumann "On the Principles of Large Scale Computing Machines", John von Neumann, Collected Works, Vol. V, p. 1, Pergamon Press, 1963.
2. G. Birkhoff, "Numerical Fluid Dynamics", SIAM Review 25, 1, (1983).
3. N. Zabusky, "Computational Synergetics", Physics Today, p. 36, July 1984.
4. Woodward, P.R. and Colella, P., "The Numerical Simulation of Two-Dimensional Fluid Flow with Strong Shocks", J. Comp. Phys. 54, p. 115, 1984.
5. Winkler, K.-H.A., Norman, M.L., and Mihalas, D., "Implicit Adaptive-Grid Radiation Hydrodynamics", in Computational Techniques: Multiple Time-Scale Problems, Ed. J.U. Brackbill and B.I. Cohen, Academic Press, 1985.
6. Winkler, K.-H.A., Norman, M.L., and Mihalas, D., "Adaptive-Mesh Radiation Hydrodynamics - I. The Radiation Transport Equation in a Completely Adaptive Coordinate System", J.Q.S.R.T., 31, p.473, 1984.